Rich variable description with Schema.org

(ESIP “schema.org” Semantic Cluster ---https://wiki.esipfed.org/Schema.org_Cluster)

Goal: Define best practices for use of so:*variableMeasured* property with so:Dataset Type

Why?

- **Support more efficient discovery of data**
- **Help users evaluate data and determine fitness for use**

Schema.org Approach

The “**variableMeasured**” property can document variables included in a “**Dataset**”… as text or objects that include an **identifier**, and a structured description using **so:PropertyValue**, or a simple so:**Text** value.

See detailed discussion document in GitHub

Related Issue and discussion in Science On Schema.org github

- Example JSON-LD code for each case is included on the following sheet.
- “so:” is used here as the abbreviation for “https://schema.org/” to conserve space.

In a Nutshell:

Minimally

- Provide Variable name, with text description
- Provide Unique identifier for Variable: improve interoperability, avoid ambiguity in identifying the variable.
- Support Indexing by Google Dataset Search if expressed as JSON-LD in <HEAD>

Ideally

Provide information about:

- Measurement technique- sampling, analytical method, data processing
- Data quality- e.g. precision, accuracy, validation procedures (not addressed here)
- Value range in data:
- Units of measure
- Data type- e.g. simple literals, links, structured objects, binary objects (image, audio, video).
- Reference established Vocabulary/Ontology
- Observation context – for the entire Dataset, or to a specific variable.

These context properties are likely to be quite domain specific. Examples:

- biome (e.g. arctic tundra) where the dataset was collected
- habitat (e.g. thermokarst) where the dataset was collected;
- the feature that was sampled (e.g.thaw lake)
- material that was sampled (talik).

Recommendations

- Use of only a simple so:Text value with *so:variableMeasured* is NOT RECOMMENDED

Basic:

Use PropertyValue *name* and *description* for text description; use *propertyID* to provide a resolvable identifier for the variable.

Notes on *propertyID*

The *propertyID* value can be an array, recognizing that there might be identifiers for variables at different conceptual levels, e.g. conceptual, representation/logical, or instance/implementation, or that the property concept might have identifiers in vocabularies used by different communities. *propertyID* values should include at least one http URI that can be resolved on the Web to yield an in-depth property description such as that included with CF names, EnvO, or a Scientific Variables Ontology, e.g. http://purl.obolibrary.org/obo/ENVO_04000002, ideally including a machine-readable representation. Clients that recognize the *propertyID* identifier, or extract useful information from its representation, can better understand what the **PropertyValue** represents.

Simple Numeric Data:

For variables with numeric values, additional properties of **so:PropertyValue** defined by schema.org should be used to provide a more complete description useful to support evaluation of a dataset for an intended use

unitText. A string that identifies a unit of measurement that applies to all values for this variable.

unitCode. Value is expected to be TEXT or URL. We recommend providing an HTTP URI that identifies a unit of measure from a vocabulary accessible on the web. The QUDT unit vocabulary (http://qudt.org/vocab/unit) provides an extensive set of registered units of measure that can be used.

minValue. If the value for the variable is numeric, this is the minimum value that occurs in the dataset. Not useful for other value types.

maxValue. If the value for the variable is numeric, this is the maximum value that occurs in the dataset. Not useful for other value types.

measurementTechnique. A text description of the measurement method used to determine values for this variable. If standard measurement protocols are defined and registered, these can be identified via http URI's.

Variables with non-numeric values:

Use the **Quantity, Units of Measure, Dimensions and Types (QUDT) ontology (http://qudt.org/)** *qudt:dataType* as a property on **so:PropertyValue** to specify the kind of data value for that property in the described dataset. The qudt schema does not constrain the domain or range of the *qudt:dataType* property. XML datatypes are recommended to populate the *qudt:dataType* property for simple literal values. Schema.org also provides Types for describing: Person, Place, Event, Date, DateTime and other common non-numeric Types that might appear as values in *so:Dataset so:variableMeasured* columns.

Variable value range is controlled vocabulary:

Options:

- Use *qudt:Enumeration* as the data type, with its content to specify a controlled vocabulary data type and range for a variable.
- Identify a controlled vocabulary as an additional *qudt:dataType* using a URI. The intention here is that the URI can be dereferenced to obtain the *qudt:Enumeration* object that defines the vocabulary elements (among other possible representations).

Variable represented by a dimensioned set of values (grid, coverage, time series, data cube)

A variable might be represented as a function of one or more dimensions. Examples:

- time series of water levels in a well;
- geospatial grid representing magnetic field intensity

A data cube structure can be implemented in various ways:

- Measured values regularly spaced along each dimension (as in many satellite imagery or time series types) in which case the dimension would be characterized by a start and end value and sample spacing.
- Dimension coordinate values might be associated with each measured value to account for irregular measured value spacing.
- There are several approaches to representing multiple measure values at each sampled location.

We do not propose how to represent the sampling points along the various dimension, only the basic value types and their semantics. Detailed description of the cube structure can be included in the dataset description or using an external vocabulary.

Recommended *qudt:dataType* for container **so:Dataset**. **qudt:MultiDimensionalDataFormatType**.

For child so:valueReference/soPropertyValue representing Dimensions use: **"propertyID": "http://purl.org/linked-data/cube#measureDimension"**

For child so:valueReference/soPropertyValue representing Dimensions use: **"propertyID": "http://purl.org/linked-data/cube#measure"**

Variable description using an external vocabulary:

If a variable Type is well-described in some external vocabulary, such as SWEET, SVO, SSN, DDI, EnvO etc., its URI reference can be given as the ***so:propertyID*** of the **PropertyValue** instance. If this approach is used, the **so:PropertyValue/*so:description*** text should contain text describing the variable scope etc. The *dcat:conformsTo* property can be asserted in the **so:PropertyValue** to identify a profile used for extending the description.

Structured values

A variable in an **attribute role** provides information about one or more of the measure value variables, e.g. to specify metadata about another variable. Examples:

- a 'units' variable that specifies the units of measure for a value in a different variable,
- a 'measurement method' variable that specifies how the value in a different variable was determined.

A measure value might be represented by a **set of component measure values,** that represent vector, tensor, tuples or object graphs. The structure can be recursive. This kind of structure is typical of JSON or XML value representations. Example:

- a location variable that has latitude, longitude and spatial reference system as component variables. The reference system value might be another structure with components.

Recommended *qudt:dataType* for container **so:PropertyValue**. Child so:valueReference elements document the attributes or component measured values:

- **Dimensional Data type**: Value specifies a physical quantity and unit of measure is embedded in the value. This is handled by the so:unitCode and so:unitText properties
- **Composite Data Type**: QUDT datatypes include several subclasses of qudt:CompositeDataType that can be used specify specific structures:
  - qudt: CompositeDataStructure. The Variable value aggregates elements of possibly different types, use to represent values that are JSON or XML type objects
  - Others include qudt:TupleType, qudt:VectorType, qudt:ArrayType.

Nested so:valueReference PropertyValue elements should describe each element in the structure with a propertyID, data type, units, etc.

Variable value is a reference

For variables that are references to data objects stored elsewhere, use the qudt:ReferenceDataType. Ideally the reference should use a scheme (like http URI) that can be dereferenced to obtain the value. If the reference is not a standard deferenceable identifier, the text description should clarify how to resolve the reference.

Example JSON-LD code

ESIP “schema.org” Semantic Cluster

Use of a simple so:Text value for variableMeasured is NOT RECOMMENDED

Basic:

Use PropertyValue *name* and *description* for text description; use *propertyID* to provide a resolvable identifier for the variable.

```
"variableMeasured":
   { "@type": "PropertyValue",
     "name": "latitude",
     "description": "Latitude where water samples were
   collected; north is positive. Latitude is a …",
     "propertyID":
   "http://semanticscience.org/resource/SIO_000319”}
```

Simple Numeric Data:

```
"variableMeasured":
    { "@type": "PropertyValue",
     "name": "latitude“,
     "propertyID":
"http://semanticscience.org/resource/SIO_000319",
     "description": "Latitude where water samples were collected; north is
positive. ",
     "unitText": "decimal degrees",
     "unitCode":"http://qudt.org/vocab/unit/DEG",
     "minValue": "45.0",
     "maxValue": "15.0”,
     “measurementTechnique": “Garmin 12 GPS”    }
```

Variables with non-numeric values:

Example dataset measured variable data type:

```
"variableMeasured":
 {"@type": "PropertyValue",
  "name": “Date of experiment",
  "description": “date and time when observation was obtained",
  "propertyID": "https://www.ex-data-repo.org/dataset-parameter/20861",
   "qudt:dataType": "xsd:dateTime" },
```

Variables that contain references

```
"variableMeasured":
 {"@type": "PropertyValue",
   "name": “EarthMaterialURI",
   "propertyID":"geosciml:gbEarthMaterialDescription",
   "alternateName": “link to rock material description",
   "description": "link to structured description of rock material using GeoSciML
properties.",
   "qudt:dataType":["xsd:anyURI", "qudt:ReferenceDatatype"]
   }
```

Variable description using an external vocabulary:

Example using the SOSA/SSN vocabulary (https://www.w3.org/TR/vocab-ssn/):

```
"variableMeasured":
{ "@type": "PropertyValue",
"propertyID": "https://www.wikidata.org/wiki/Property:P5596",
"name": "Relative Humidity",
"dcat:conformsTo":"https://www.w3.org/TR/2017/REC-vocab-ssn-20171019/",
"sosa:isResultOf": {
   "@type":"sosa:Observation",
   "rdfs:comment": "Relative humidity as averaged over 15min at COPR.",
    "rdfs:label“: "Relative humidity, AVG, 15min, COPR, 06.02.2017, 3:00 PM“,
    "sosa:madeBySensor": "http://example.org/data/HUMICAP-H",
    "sosa:hasFeatureOfInterest":  "http://example.org/data/COPR_Station",
    "sosa:observedProperty":
                  “http://sweetontology.net/propFraction/RelativeHumidity",
    "sosa:usedProcedure":
  “https://www.globe.gov/documents/348614/348678/Relative+Humidity+Protocol/89f8c...  }
```

Other schema.org PropertyValue properties omitted here, but could be included.

Variable value range is controlled vocabulary:

Example encoding for a variableMeasured that is populated with a controlled vocabulary, using qudt:dataType/qudt:Enumeration to list the allowed values:

```
"variableMeasured":
 {  "@type": "PropertyValue",
  "propertyID": "http://astromat/parameters/0027",
   "name": "calcAvg",
   "description": "Value in sample data are 'Can be averaged', 'Cannot be averaged', 'It is
average’”,
   "qudt:dataType": {
     "qudt:Enumeration": {
       "qudt:element": [
         {"qudt:EnumeratedValue": {"qudt:symbol":"Can be averaged"}},
         {"qudt:EnumeratedValue": {"qudt:symbol":"Cannot be averaged"}},
         {"qudt:EnumeratedValue": {"qudt:symbol":"It is average"}}       ]
       }
   }  }
```

Example referencing controlled vocabulary via URI:

```
"qudt:dataType": [“qudt:Enumeration”, “https://www.astromat.org/vocab/isaverage”]
```

Structured values: Composite Data Type

```
{ "@type": "PropertyValue",
  "name": "PLSSLocation",
  "propertyID":"http://www.opengis.net/def/property/OGC/0/SamplingLocation",
  "alternateName": "US Public Land Survey System location",
  "description": "Location of sampling feature specified using PLSS grid",
  "qudt:dataType": [“qudt:TupleType", "https://www.usgs.gov/media/images/public-land-survey-system-plss"],
  "valueReference": [
    {"@type": "PropertyValue",
      "name": "PLSS_Meridians",
      "description": “N-S baseline and E-W meridian reference for TWP and RGE.",
      "qudt:dataType": "xsd:token"  },
    {"@type": "PropertyValue",
      "name": "TWP",
      "alternateName": "Township",
      "description": "Township in PLSS grid, relative to reported baseline. ",
      "qudt:dataType": "xsd:token"    },
    {"@type": "PropertyValue",
      "name": "RGE",
      "alternateName": "Range",
      "description": "Range in PLSS grid, relative to reported meridian.",
      "qudt:dataType": "xsd:token"  }
        ]
 },
```

Variable represented by a dimensioned set of values

Two approaches for discussion. One approach:

Group dimensions and measure as separate properties, with a tuple structure

```
{"@type": [ "Dataset", "qudt:MultiDimensionalDataFormat" ],
   "name": "Surface geology and geophysics grid",
…
 "variableMeasured": [
  {"@type": "PropertyValue",
   "name": "Dimensions",
   "propertyID": "http://purl.org/linked-data/cube#measureDimension",
   "description": "The dimensions for logical space in which measured values
are positioned...",
   "qudt:dataType": "http://qudt.org/schema/qudt/TupleType",
   "valueReference": [
      {"@type": "PropertyValue",
       "name": "latitude",
       "propertyID": "http://semanticscience.org/resource/latitude",
       "qudt:dataType": "xsd:decimal",
       "unitText": "decimal degree"  },
      {"@type": "PropertyValue",
       "name": "longitude",
       "propertyID": "http://semanticscience.org/resource/longitude",
       "qudt:dataType": "xsd:decimal",
       "unitText": "decimal degree"}
    ]
  },
  {"@type": "PropertyValue",
   "name": “measure value", qudt:DimensionalDatatype
   "propertyID": "http://purl.org/linked-data/cube#measure",
   "description": "tuple with magnetic field intensity, g value, observed
outcrop rock type, and elevation",
   "qudt:dataType": "http://qudt.org/schema/qudt/TupleType",
   "valueReference": [
     {"@type": "PropertyValue",
       "name": "mag",
       "alternateName": "magnetic field intensity",
       "propertyID": "http://ex.org/resource/magneticFieldIntensity",
        "qudt:dataType": "xsd:decimal",
        "unitText": "amperes per metre" },
…. Other measure properties omitted
}
```

Alternate approach:

Use propertyID to differentiate dimensions and measures

```
…
"variableMeasured": [
    {   "@type": "PropertyValue",
       "name": "latitude",
       "propertyID": [
          "http://purl.org/linked-data/cube#measureDimension",
          "http://ex.org/resource/latitude"        ],
       "qudt:dataType": "xsd:decimal",
       "unitText": "decimal degree"       },
    {  "@type": "PropertyValue",
       "name": "longitude",
       "propertyID": [
          "http://purl.org/linked-data/cube#measureDimension",
          "http://ex.org/resource/longitude"          ],
       "qudt:dataType": "xsd:decimal",
       "unitText": "decimal degree"       },
    {   "@type": "PropertyValue",
       "name": "mag",
       "alternateName": "magnetic field intensity",
       "propertyID": [
          "http://purl.org/linked-data/cube#measure",
          "http://ex.org/resource/magneticFieldIntensity"         ],
       "qudt:dataType": "xsd:decimal",
       "unitText": "amperes per metre"       },
…. Other measure properties omitted
}
```

Background Notes

Science on Schema.org

- Web-friendly architecture for disseminating metadata about **datasets**

The Challenge:

- Schema.org (SDO) originated for marking up web pages for commercial activity
  - E.g. Concerts, Books, Movies, Music Recordings, Recipes, TVSeries….
- Vocabulary is very large, tricky to navigate for beginners
- Usage is very loosely constrained.
  - Very flexible, handy for people looking at search results
  - Not interoperable for machine agents

Solution:

- Develop recommendations for science community to provide consistent, machine-actionable metadata to enable more efficient data discovery

Components:

- Schema.org JSON-LD embedded in web pages
- Sitemap that provides URLs for web pages containing metadata

What is a sitemap?

- A file that is placed in a root directory for a website
  - Contains links for pages to index in that website
  - Links have time stamp for most recent update of content at that location

- Linked from 'robots.txt'– another file in the website root directory that contains directives for (well-behaved…) web-crawlers
  - Contains instructions for what should and should not be indexed on that website
  - Can point to sitemap to guide crawlers

What is schema.org (SDO)?

- In the beginning… an RDF vocabulary of entities and properties for semantic 'enrichment' of web pages.
- Developed by major search providers– Google, Bing, Yahoo; used to enhance presentation and functionality of search results
- Dataset Search– use SDO vocabulary to document data (metadata!)
  - Embed metadata in landing pages typically offered by dataproviders
  - Sitemaps (see box to left) guide search crawlers to web pages that contain metadata
  - Metadata content used to populate indexes that support search, e.g. Google Dataset Search, EarthCube GeoCODES
  - See https://developers.google.com/search/docs/data-types/dataset

Current Cluster activities

Dataset Recommendations v1.1

Issues for upcoming releases:

v1.2 (target January 2021)

- Add provenance for data with isBasedOn links
- Add SHACL shape to validate consistent schema.org namespace.
- Recommendation on specifying dateModified for Datasets
- Recommendation for GeoShape bounding box format
- Documentation on referencing a short DOI

V1.3

- Documentation on usage of "citation"
- Add OWL-Time Extension guidance
- Recommendation for indicating authoritative copy of dataset
- Flesh out a rich variable description of a dataset
  - ○Representing ontological terms representing observation types of variableMeasured

Google's definition of dataset:

From https://developers.google.com/search/docs/data-types/dataset

Here are some examples of what can qualify as a dataset:

•A table or a CSV file with some data

•An organized collection of tables

•A file in a proprietary format that contains data

•A collection of files that together constitute some meaningful dataset

•A structured object with data in some other format that you might want to load into a special tool for processing

•Images capturing data

•Files relating to machine learning, such as trained parameters or neural network structure definitions

•**Anything that looks like a dataset to you**